
Short Tutorial on VXI/MXI

Ron Wolfe

The purpose of this application note is to help you gain an understanding of VXI and MXI concepts. This application note is divided into two tutorial sections. The first section discusses VXI and the second section discusses MXI.

VXI Tutorial

This section contains an overall introduction to VXI (VMEbus eXtensions for Instrumentation).

What is VXI?

VXIbus is an exciting and fast-growing platform for instrumentation systems. The VXI Consortium was formed in 1987 with a charter of defining a multivendor instrument-on-a-card standard. Since that time, the Consortium has defined system-level components required for hardware interoperability. The IEEE officially adopted the VXI specification, IEEE 1155, in March 1993. The VXI*plug&play* Systems Alliance, founded in September 1993, sought a higher level of system standardization to cover all VXI system components. By focusing on software standardization, the alliance defined standards to make VXI systems easy to integrate and use while maintaining multivendor software interoperability. With the success of multivendor standards and solid technical specifications, VXI is backed by more than 250 vendors, with more than 1000 products available. The success of VXI as an open, multivendor platform is a testament to the value of multivendor standards, and has made VXI the platform of choice for open instrumentation systems.

VXI is used in many different applications ranging from test and measurement and ATE, to data acquisition and analysis in both research and industrial automation. Although some VXI systems today are purely VXI, many users are migrating to VXI by integrating it into existing systems consisting of GPIB instruments, VME cards, or plug-in data acquisition (DAQ) boards. You can control a VXI system with a remote general-purpose computer using the high-speed Multisystem eXtension Interface (MXI) bus interface or GPIB. You can also embed a computer into a VXI chassis and control the system directly. Whatever your system configuration needs may be, VXI offers the flexibility and performance to take on today's most challenging applications.

The Need for VXIbus

The demand for an industry-standard instrument-on-a-card architecture has been driven by the need for physical size reduction of rack-and-stack instrumentation systems, tighter timing and synchronization between multiple instruments, and faster transfer rates than the 1 Mbytes/s rate of the 8-bit GPIB. The modular form factor, high bandwidth, and commercial success of the VMEbus made it particularly attractive as an instrumentation platform. The tremendous popularity of GPIB also made it attractive as a model for device communication and instrument control protocols. The VXIbus specification adds the standards necessary to combine the VMEbus with GPIB to create a new, modular instrumentation platform that can meet the needs of future instrumentation applications.

Product and company names are trademarks or trade names of their respective companies.

VXI brings the following benefits to instrumentation users:

- Open, multivendor standards maximize flexibility and minimize obsolescence
- Increased system throughput reduces test time and/or increases capabilities
- Smaller size and higher density reduce floor space, enhance mobility or portability, and give close proximity to devices(s) being tested or controlled
- More precise timing and synchronization improve measurement capability
- Standardized *VXIplug&play* software eases system configuration, programming, and integration
- Modular, rugged design improves reliability, increases mean-time between failure (MTBF), and decreases mean-time to repair (MTTR)

Members of the VXIbus Consortium and the *VXIplug&play* Systems Alliance have combined their expertise to develop technically sound standards for both hardware and software, bringing the entire industry into a new generation of easy-to-use instrumentation.

The Value of Open Industry Standards

The baseline VXI hardware specifications are a mandate for interoperability between hardware products from different vendors. These specifications cover mechanical and environmental requirements such as module sizes, mainframe and module cooling, and EMC compatibility between modules, as well as automated system initialization and backplane communication protocols. The *VXIplug&play* Systems Alliance builds on these baseline specifications to address the system as a whole with the goal of having the end-user up and running in “Five minutes or less.” Building a system based on open industry standards means that you choose components for your system based on your requirements, regardless of vendor. Open standards also ensure that once your system is built, your investment will continue to pay dividends well into the future.

Both the VXI Consortium and the *VXIplug&play* Systems Alliance remain strong, active organizations committed to maintaining VXI as an open, multivendor technology and increasing its ease of use and end-user success. In fact, many of the largest instrument suppliers in the world are members of both organizations, including National Instruments, GenRad, Hewlett-Packard, Racal Instruments, and Tektronix. With *VXIplug&play*, you are assured that components from different vendors will work reliably in the same system. Members of the VXI Consortium and the *VXIplug&play* Systems Alliance have combined their expertise to develop technically sound standards for both hardware and software, bringing the entire industry into a new generation of instrumentation – a generation that stresses ease of use and open systems without sacrificing flexibility or performance.

VXIbus Mechanical Configuration

Physically, a VXIbus system consists of a mainframe chassis that has the physical mounting and backplane connections for plug-in modules, as shown in Figure 1. The VXIbus uses the industry-standard IEEE-1014 VMEbus as a base architecture to build upon. As shown in Figure 2, VXI uses the full 32-bit VME architecture, but adds two board sizes and one connector. The P1 connector and the center row of the P2 connector are retained exactly as defined by the VME specification. The VME user-definable pins on the P2 connector and the additional pins on P3, the third VXI connector, implement instrumentation signals between plug-in modules directly on the backplane.

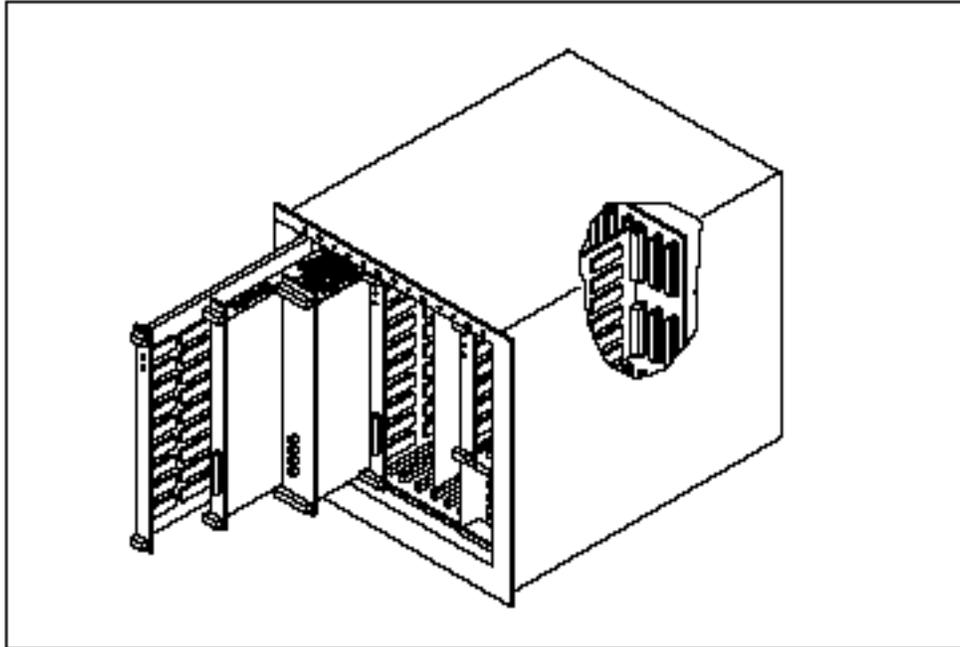


Figure 1. A VXIbus System

The VXIbus specification includes packaging requirements, electromagnetic compatibility, power distribution, cooling and air flow for VXIbus mainframes and plug-in modules. The modules are installed in the mainframe slots. LEDs, switches, test points, and I/O connections are accessible from the module front panel.

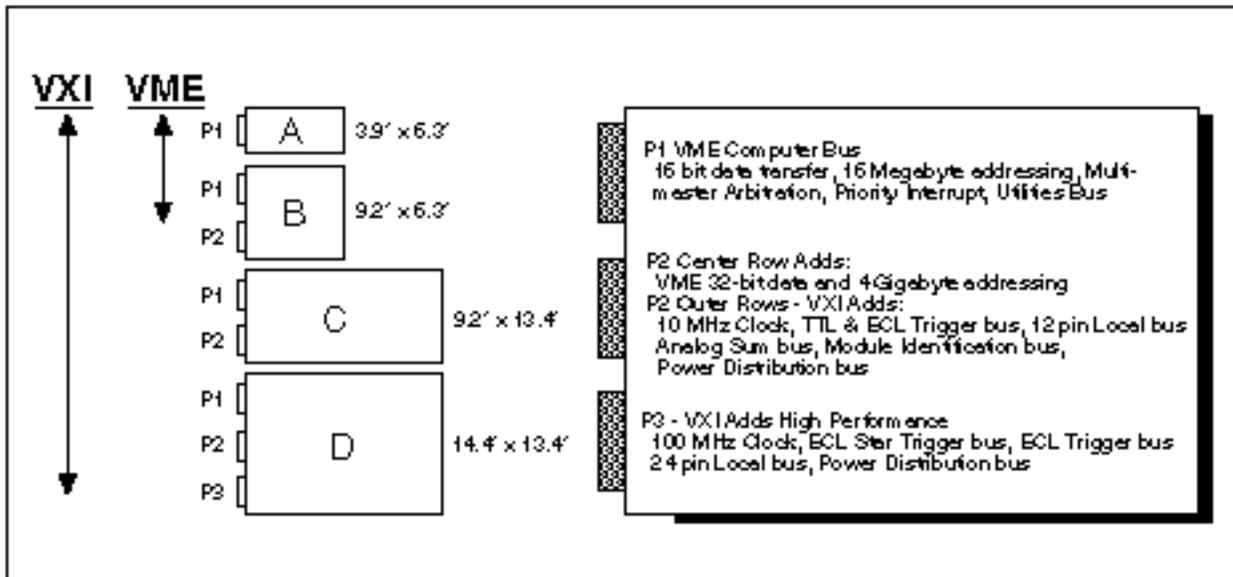


Figure 2. VXI Module Sizes

Module and Mainframe Cooling

Airflow direction is from bottom (P3) to top (P1). Cooling requirements must be established for all modules and included in product specifications. These requirements must include an operating point of minimum airflow requirement. Mainframe suppliers must also provide similar information for their mainframes.

EMC and Noise

The addition of a new module to a VXIbus system must not degrade the performance of any other module. The VXIbus specification includes near-field radiation and susceptibility requirements, which prevent one module from interfering with the operation of other modules. To help meet these requirements, the VXIbus module width was increased from the 0.8 in. VME requirement to 1.2 in., so that there is enough room for the module to be completely enclosed in a metal case for shielding. The metal cases connect to backplane grounds. Thus, you can use existing VME boards in a VXIbus chassis, but not vice versa.

The VXIbus specification also has conducted-emissions and susceptibility requirements, which prevent any power supply noise from affecting the performance of a module. For far-field radiated emissions such as FCC and VDE, each module must not contribute more than its share of the total. For example, in a mainframe that holds 13 modules, each module must not contribute more than 1/13 of the allowed total. Because of the desire for extremely precise time coupling between modules using the backplane, it is necessary to minimize the noise and crosstalk on the backplane clock and trigger signal lines. The backplane is required to be a single, monolithic board across any one slot. The VXIbus specification has a tutorial section on how to design a backplane for low noise and high signal integrity.

Hardware Registers

VXI modules must have a specific set of registers located at specific addresses, as shown in Figure 3. The upper 16 KB of the 64 KB A16 address space are reserved for VXIbus devices. Each VXI device has an 8-bit logical address that specifies where its registers are located in this address space. A single VXI system can have up to 256 VXI devices. The logical address of a VXI device, which can be manually set or automatically configured by the system at startup, is analogous to the GPIB address of a GPIB device.

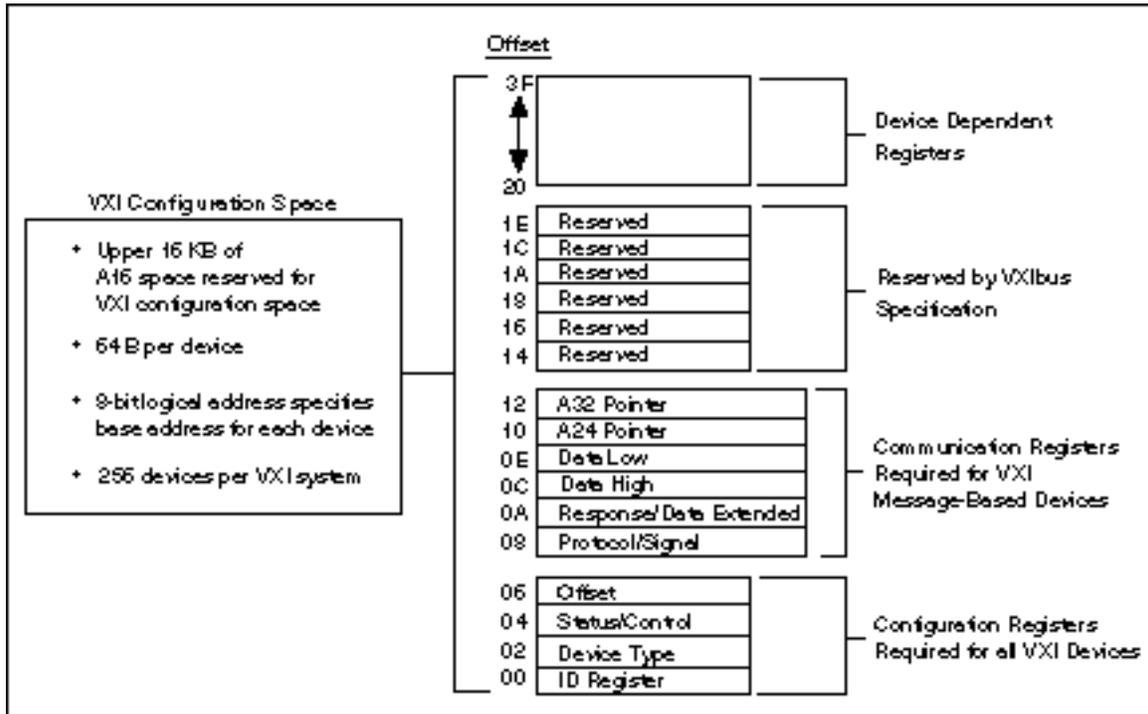


Figure 3. VXI Configuration Registers

Register-Based Devices

Because of the VXI configuration registers, which are required for all VXI devices, the system can identify each VXI device, its type, model and manufacturer, address space, and memory requirements. VXIbus devices with only this minimum level of capability are called Register-Based devices. With this common set of configuration registers, the centralized Resource Manager (RM), essentially a software module, can perform automatic system and memory configuration when the system is initialized.

Message-Based Communication

In addition to Register-Based devices, the VXIbus specification also defines Message-Based devices, which are required to have communication registers and configuration registers. All Message-Based VXIbus devices, regardless of the manufacturer, can communicate at a minimum level using the VXI-specified Word Serial Protocol. When minimum communication is possible, higher-performance communication channels, such as shared-memory channels, can be established to take advantage of the VXIbus bandwidth capabilities.

Word Serial Protocol

The VXIbus Word Serial Protocol is functionally very similar to the IEEE-488 protocol, which transfers data messages to and from devices one byte (or word) at a time. Thus, VXI Message-Based devices communicate in a fashion very similar to IEEE-488 instruments. In general, Message-Based devices typically contain some level of local intelligence that uses or requires a high level of communication.

All VXI Message-Based devices are required to use the Word Serial Protocol to communicate in a standard way. The protocol is called *word serial*, because if you want to communicate with a Message-Based device, you do so by writing and reading 16-bit words one at a time to and from the Data In (write Data Low) and Data Out (read Data Low) hardware registers located on the device itself. Word Serial communication is paced by the bits in the response register of the device, indicating whether the Data In register is empty and whether the Data Out register is full. This operation is very similar to Universal Asynchronous Receiver Transmitter (UART) on a serial port.

Commander/Servant Hierarchies

The VXIbus defines a Commander/Servant communication protocol so you can construct hierarchical systems using conceptual layers of VXI devices. This structure is like an inverted tree. A Commander is any device in the hierarchy with one or more associated lower-level devices, or Servants. A Servant is any device in the subtree of a Commander. A device can be both a Commander and a Servant in a multiple-level hierarchy.

A Commander has exclusive control of the communication and configuration registers of its immediate Servants (one or more). Any VXI module has one and only one Commander. Commanders communicate with Servants through the communication registers of the Servants using the Word Serial Protocol if the Servant is a Message-Based device, or by device-specific register manipulation if the Servant is a Register-Based device. Servants communicate with their Commander by responding to the Word Serial commands and queries from their Commander through the Word Serial protocol if they are Message-Based, or by device-specific register status if they are Register-Based.

Interrupts and Asynchronous Events

Servants can communicate asynchronous status and events to their Commander through hardware interrupts or by writing specific messages (signals) directly to their Commander's hardware Signal Register. Nonbusmaster devices always transmit such information via interrupts, whereas devices that have busmaster capability can either use interrupts or send signals. Some Commanders can receive signals only, whereas others might be only interrupt handlers.

The VXIbus specification contains defined Word Serial commands so that a Commander can understand the capabilities of its Message-Based Servants and configure them to generate interrupts or signals in a particular way. For example, a Commander can instruct its Servants to use a particular interrupt line, to send signals rather than generate interrupts, or configure the reporting of only certain status or error conditions.

Although the Word Serial Protocol is reserved for Commander/Servant communications, peer-to-peer communication between two VXI devices can be established through a specified shared-memory protocol or by simply writing specific messages directly to the signal register of the device.

Slot 0 and the Resource Manager

The leftmost slot of a VXI chassis has special system resources such as backplane clocks, configuration signals, and synchronization (trigger) signals and therefore must be occupied by a device with VXI “Slot 0” capabilities. The VXI Resource Manager (RM) function, essentially a software module, can reside on any VXI module or even on an external computer. The RM, in combination with the Slot 0 device, identifies each device in the system, assigns logical addresses, memory configurations, and establishes Commander/Servant hierarchies using the Word Serial Protocol to grant Servants to the Commanders in the system. After establishing the Commander/Servant hierarchy, the RM issues the Begin Normal Operation Word Serial command to all top-level Commanders. During normal system operation, the RM may also halt the system and/or remap the hierarchy if necessary.

Three Ways to Control a VXI System

System configuration is divided into three categories. The first type of VXI system consists of a VXI mainframe linked to an external controller via the GPIB. The controller talks across the GPIB to a GPIB-VXI interface module installed inside the VXI mainframe. The GPIB-VXI interface transparently translates the GPIB protocol to and from the VXI Word Serial protocol.

The second configuration involves a VXI-based embedded computer. The embedded computer is a VXI module that resides inside the VXI mainframe and connects directly to the VXI backplane. This configuration offers the smallest physical size for a VXI system as well as performance benefits due to direct connection to the VXI backplane.

The third configuration uses a high-speed MXIbus link from an external computer to control the VXI backplane. The external computer operates as though it is embedded directly inside the VXI mainframe. This configuration is functionally equivalent to the embedded method, except that it has the flexibility for use with a wide variety of computers and workstations.

VXI Bus Interface Software

One of the most important considerations when selecting a VXI system is software. Software is the key to developing successful systems based on the VXIbus. There are many programming languages, operating systems, and application development environments (ADE) to choose from when building a VXI system. It is important to make the right decisions to realize all of the advantages that VXI has to offer, while minimizing your development costs now and in the future.

Your software decisions not only affect overall system performance and system capability, but also development time and productivity. You should choose tools that have complete debugging capability and that work with the most popular operating systems and programming languages. If you choose to program your VXI system using a standard language such as C, C++, Basic, ADA, or ATLAS, you should realize that standard programming languages do not come with built-in VXI capability. Rather, VXI capability is added through a VXI bus interface software library. This software component is very important, because it affects the choice of VXI computer hardware, operating system, programming language, and ADE.

Industry-Wide Software Standards

As a step toward industry-wide software compatibility, the *VXIplug&play* alliance developed one specification for I/O software – the Virtual Instrument System Architecture, or VISA. The VISA specification, VPP-4.1, defines a next-generation I/O software standard not only for VXI, but also for GPIB and serial interfaces. With the VISA standard endorsed by over 50 of the largest instrumentation companies in the industry including Tektronix, Hewlett-Packard, and National Instruments, VISA unifies the industry by facilitating the development of interoperable and reusable software components able to stand the test of time. Before VISA, there were many different commercial implementations of I/O software for VXI, GPIB, and serial interfaces; however, none of these I/O software products were standardized or interoperable.

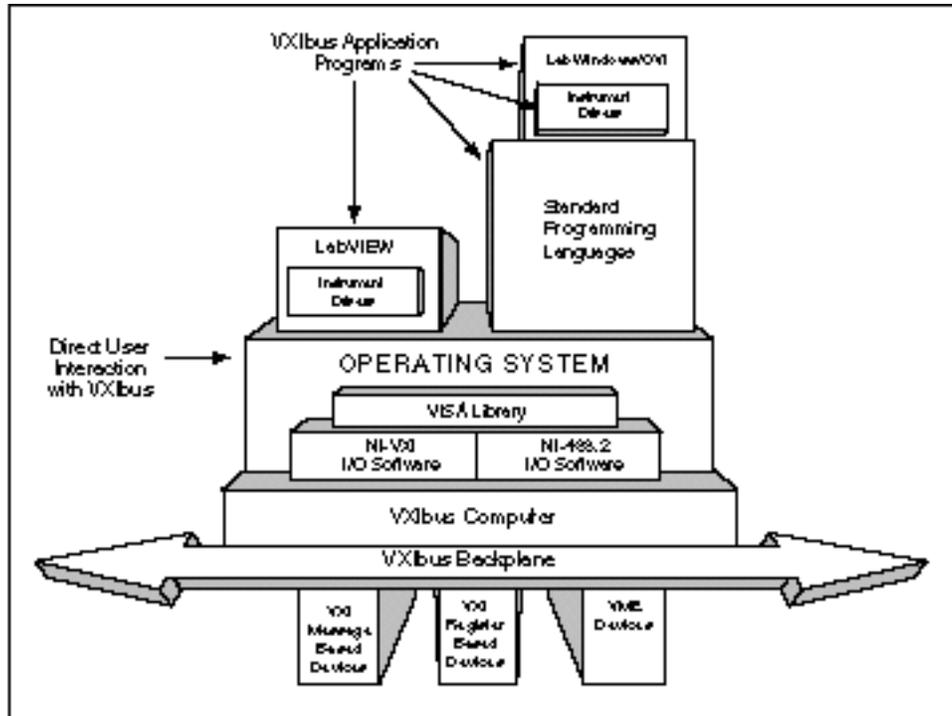


Figure 4. VXIbus System Software Components

The VISA standard lays the foundation and provides a unified migration path for industry-wide software compatibility. One of the most notable benefits of VISA is its ability to significantly reduce the time and effort involved in programming different I/O interfaces. Instead of using a different Application Programmers Interface (API) devoted to each interface bus, you can use the VISA API regardless of whether your system is controlled by GPIB, VXI, or a GPIB-VXI.

With the vast number of choices in instrumentation and software now available, most users do not want to be locked into a specific vendor for their systems. Instead, they would prefer the freedom to select the best instruments and software available from multiple vendors and have it all work together with minimal effort. The IEEE 488.1 and IEEE 488.2 standards (for GPIB) and the IEEE 1155 standard (for VXI) ensured that the hardware would be interoperable, but this approach was not taken for the software. Therefore, the ideal new driver architecture should be a standard adopted by as many of the major vendors as possible. Then you could be assured that any code written for your instrument is portable across controller vendors as well as operating systems. This is exactly what the *VXIplug&play* Systems Alliance has done with VISA.

MXI Tutorial

This section contains an overall introduction to MXI.

MXIbus Overview

The MXI bus is a powerful, high-speed communication link that interconnects devices using a flexible cabling scheme. Derived from the VMEbus, MXI provides a high-performance way of controlling VXI systems using commercially available desktop computers and workstations. National Instruments developed and published the MXI specification and released it as an open industry standard in 1989. In 1995, National Instruments introduced MXI-2, which offers even higher performance.

A MXIbus system configuration combines the performance benefits of a custom embedded VXI computer with the flexibility and availability of general-purpose computers. The MXIbus system configuration uses the high-speed MXIbus cable to connect an external computer directly to the VXI backplane. With the MXIbus, you can locate the computer directly next to the VXI mainframe, or up to 20 meters away. Using the MXIbus, you can easily add other VXI mainframes, and use the plug-in slots in the external computer for GPIB-control, plug-in DAQ boards, or other peripheral adapter cards.

For instrument control, MXI complements high-speed platforms such as PCI by harnessing their high-throughput potential. PCI-based desktop PCs compete with the most advanced computer workstations to provide a low-cost platform that delivers superior performance. You can use low-cost desktop computers to control sophisticated VXI instrumentation without sacrificing performance or control. More importantly, as new desktop computers incorporate the latest technology including faster, more capable microprocessors and RAM, you can easily upgrade your VXI system as these newer and faster computers emerge to immediately reap increased VXI performance gains. Thus, a PCI-based MXI-2 solution such as our VXI-PCI8000 gives you excellent performance now with headroom for the future.

A New Generation of VXI Connectivity

Many VXI users migrate from GPIB-based systems. As a result, the National Instruments GPIB-VXI is a popular way to control VXI instruments from a GPIB controller. An increasingly popular way to control VXI, however, is to use a custom VXI computer that plugs directly into the VXI mainframe, such as the National Instruments VXIpc[™] and VXIcpu[™] Series of embedded VXI computers. This *embedded* approach is technically attractive because the computer communicates directly with the VXIbus and is tightly coupled to the instruments.

Although an embedded computer is very powerful, custom VXI computers cannot possibly keep pace with the general-purpose computer market. In the last decade, specialized instrument controllers have rapidly declined. General-purpose PCs and workstations, with their vast array of software and accessories, have revolutionized our industry. By using general-purpose computers, the instrumentation industry directly benefits from the billions of R&D dollars spent each year in the general computer market.

Most VXI users would prefer to use an industry-standard computer provided by a computer vendor rather than a VXI-specific computer provided by an instrument vendor. In fact, for VXI to truly become the platform for the next millennium, it must align itself with the powerful general computer market. Then VXI can take advantage of the billions of dollars being spent and bring this investment to bear on the needs of the instrumentation community. VXI must be able to take full advantage of industry-standard PCs with PCI, EISA, and ISA, as well as workstations from Sun, HP, and others. VXI also must have a transparent mechanism for extending to multiple mainframes, and a way to accommodate instruments that cannot physically fit on a VXI module. MXIbus meets each of these needs.

The Need for MXIbus

Today's market demands that you add value to our Test & Measurement systems. You need modular testing systems that can evolve with technological innovations in our industry. You want increased data throughput and the utmost in computing power; you want flexible, high-speed connectivity between multiple VXI/VME mainframes; and you want to be able to keep up with innovations in PC and workstation technology. Today, sophisticated I/O architectures such as PCI are accelerating data throughput – who knows what tomorrow may hold. How can you take these benefits both now, and in the future? We believe that the answer is MXI.

MXI provides you with a solution that combines the performance benefits of an embedded VXI computer with the flexibility of a general purpose desktop computer. Our VXI-PCI8000 controller and our next generation MXI-2 provides you with an ultra high-performance VXI connectivity solution that can meet your needs both today and well into the future. Although traditional connectivity solutions have proved to be very effective, they also have proved to be the bottleneck in VXI test systems because the software protocol overhead associated with these methods significantly reduced the achievable throughput on the link. Using MXI, this bottleneck is eliminated altogether because MXI devices are connected at the hardware level by mapping each physically separate system into a shared memory space. Physically separate devices transparently share resources through simple reads and writes to the appropriate address in memory. Our next generation MXI-2 products enhance VXI connectivity by defining a single memory-mapped backplane-on-a-bus that can transparently extend bus-level I/O, VXI triggering, interrupts, and systems clocks between systems. You can now use a single cable to conveniently share trigger and timing information between mainframes in a multiple mainframe configuration. The MXI 2.0 specification also defines a synchronous data transfer method that increases MXIbus throughput for block data transfers. From a system standpoint, this means that MXI throughput rates can easily keep up with the data rates of high-performance computers, peripherals, and instrumentation. From a user standpoint, this translates to increased performance and reduced time to test. By choosing a PC-based MXI approach, you are choosing to add value to your VXI instrumentation systems by leveraging technologies that make sense from both a cost and performance perspective.

MXIbus Applications

You can use MXIbus for a variety of applications. You can interface industry-standard desktop computers to VXIbus or VMEbus; you can create multiple chassis configurations using our VXI-MXI or VME-MXI extenders; and you can integrate VXI and VME chassis into the same test system.

Figures 5 and 6 show two common configurations with MXIbus.

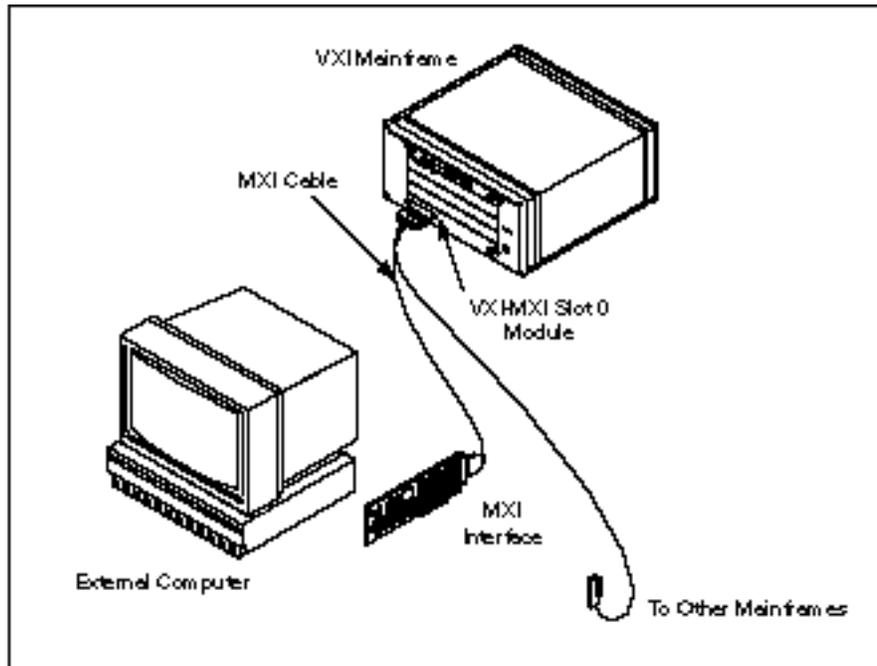


Figure 5. PC Using MXI to Control a VXIbus System

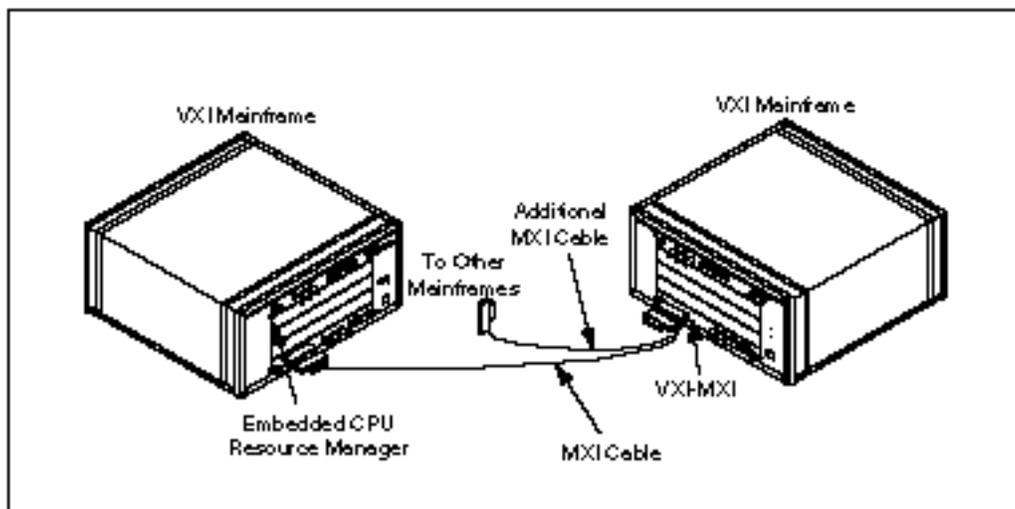


Figure 6. MXI Used for Multiple Mainframe VXI System

How Does MXIbus Work?

MXIbus is a general purpose, 32-bit multimaster system bus on a cable. MXI interconnects multiple devices using a flexible cabling method similar to GPIB, but uses a hardware memory-mapped communication scheme that eliminates the software overhead. MXI devices can directly access each other's resources by performing simple read and writes to appropriate address locations. The new MXI-2 standard expands on the MXI-1 standard by exporting all VXI backplane signals such as VXI-defined trigger lines, interrupt lines and system clocks, in addition to the standard MXIbus signals directly to the cabled bus. MXI-2 users can accomplish critical timing and synchronization tasks between up to eight, daisy-chained MXI devices.

MXI device connectivity is accomplished at the hardware level. The MXI cable serves as a transparent link that interconnects multiple MXI devices. These devices are interlaced by mapping together portions of their individual address spaces so that a system composed of multiple devices behaves as a single system with a shared address space. Figure 7 shows the MXIbus hardware memory-mapped communication. The immediate benefit of this approach is increased data throughput due to the absence of software overhead.

Each MXIbus hardware interface has *address window* circuitry that detects internal (local) bus cycles that map out to the MXIbus. In addition, this circuitry also detects external (remote) MXIbus cycles of connected devices whose addresses map into the shared memory space of the overall system. When a hardware write or read occurs with an address that maps across MXI, the MXI hardware interlocks the bus cycle between the devices via the MXIbus. This hardware scheme is the same as that used by embedded VXI controllers.

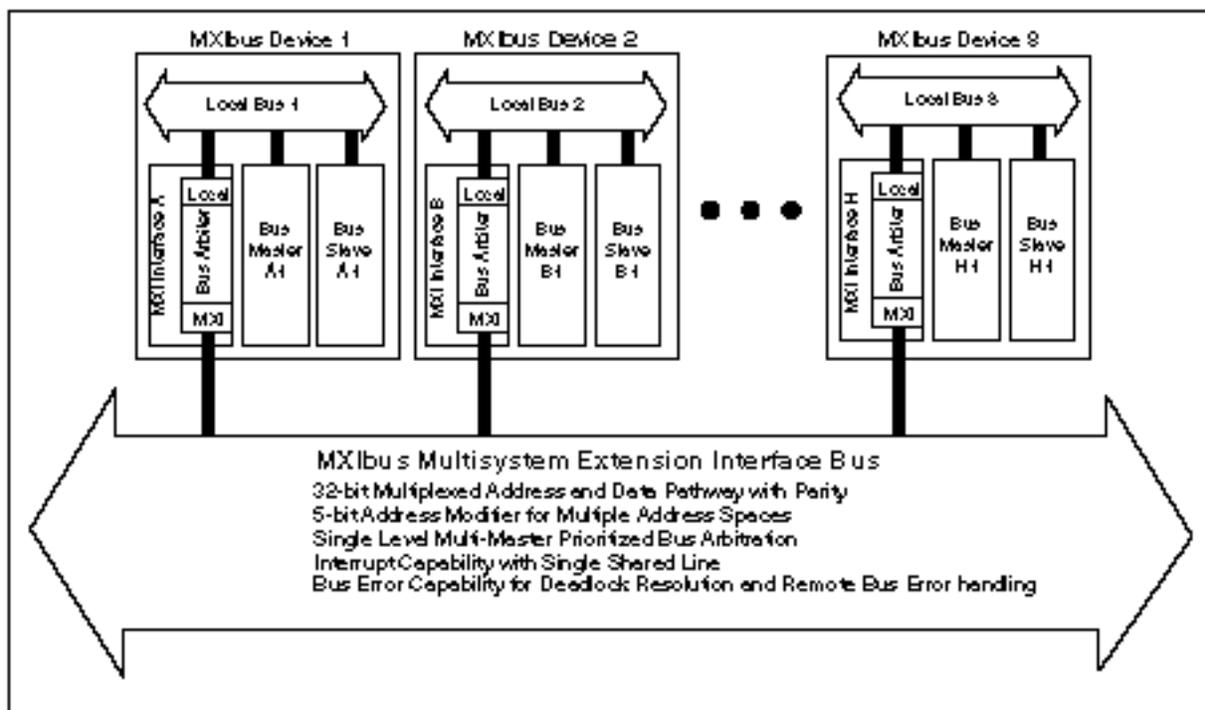


Figure 7. MXIbus Hardware Memory-Mapped Communication

MXIbus Signals

MXIbus signals include 32 multiplexed address and data lines with parity, address modifiers for multiple address spaces, single-level multimaster prioritized bus arbitration, a single interrupt line, a bus error line for handling timeouts and deadlock conditions, and handshake lines for asynchronous operation. Data transfers of 8, 16, and 32 bits are possible, as well as invisible read/write operations and integrated block-mode transfers. With synchronous MXI, the MXI-2 product line can achieve burst data rates as high as 33 Mbytes/s, and sustained throughput rates exceeding 20 Mbytes/s, regardless of the length of the MXI-2 cable

MXIbus Cables

A single MXI cable can be any length up to 20 m. Up to eight MXI devices can be daisy-chained on a single MXI cable length. If multiple MXI devices are daisy-chained together, the total cable distance must be no more than 20 m. The MXI-1 cable is a flexible, round cable similar to a GPIB cable (about 0.6 in. in diameter). Internally there are 48 single-ended, twisted-pair signal lines. MXI-2 features an improved cabling scheme that uses a single double-shielded cable between all devices, and a single high-density, high reliability 144-pin connector per device. In this fashion, all MXI-2 devices share not only the MXIbus itself, but also the VXI-defined trigger lines, interrupt lines, systems clocks, and other signals that were available on MXI-1 products as an optional second connector and cable (INTX). MXI-1 products use an MXI-1 cable between devices, and an optional INTX cable to share trigger/timing information between mainframes in a multiple mainframe configuration. MXI-2 eliminates the need for an additional INTX cable in your system. Because of the cables differences, you cannot mix MXI-1 and MXI-2 products in the same system. Both MXI-1 and MXI-2 use double shielding with an aluminum mylar shield as well as a copper braid shield to eliminate any EMI problems, and both cables meets the National Electric Code (NEC) CL2 fire safety code. The stacking depth of two daisy-chained MXI cables is approximately 3.3 in.

MXI is essentially a backplane bus in a cable. Each MXI signal line is twisted with its own ground line. All MXI signal lines are matched impedance to minimize signal skew and reflections. Stub lengths no more than 4 in. off the mainline interconnection minimize reflections due to impedance discontinuities. Termination networks, configured with onboard jumpers, are located at the first and last MXI devices to minimize reflections at the ends of cables.

MXI uses state-of-the-art, single-ended, trapezoidal bus transceivers to reduce noise crosstalk in the transmission system. Designed specifically for driving backplane bus signals, these transceivers have open-collector drivers that generate precise trapezoidal waveforms with typical rise and fall times of 9 ns. The trapezoidal shape, due to the constant rise and fall times, reduces noise coupling (crosstalk) on adjacent lines. The receiver uses a lowpass filter to remove noise and a high-speed comparator that recognizes the trapezoidal-shaped signal from the noise.

Performance Issues

MXIbus Performance

It is often difficult to understand how a performance specification for a single component relates to the overall performance of your system. In the case of MXI, it is important to understand not only the performance issues associated with the MXI link, but also the devices that communicate across the link. MXI works like an embedded computer, using a direct hardware memory-map to eliminate software overhead between your computer and the VXIbus or VMEbus. Both MXI and embedded VXI computers can use shared-memory communication protocols and direct register accesses for potentially dramatic performance improvements over GPIB. If your VXI instruments themselves do not use these capabilities, however, your system performance using MXI or an embedded computer may be no higher than a GPIB-controlled VXI system.

There are several factors to consider when comparing a MXI-equipped computer to an embedded computer. A MXI-equipped computer is functionally equivalent to an embedded computer. In fact, application software developed on a MXI computer using NI-VXI/VISA bus interface software can easily run on an embedded computer and vice versa. There are subtle hardware timing differences, but there is no dramatic performance difference due to architecture. MXI, for example, can take roughly 100 ns more to perform a single VXI read or write than an embedded computer, because the MXI signals must propagate down the MXI cable at 10 ns/m. This subtle detail is

measured in ns, and is negligible compared to the other factors that affect your system performance, such as the execution speed of your application software or your instruments.

Often, the most important performance issue to consider when evaluating a computer for your system is the performance of the processor itself. Most applications spend much more time computing, displaying, or performing disk I/O than actually performing I/O across the VXIbus or VMEbus. Current external MXI computers are over four times as fast as the fastest embedded VXI computer. In addition, because of the physical space constraints of embedded computers, external computers often have much more sophisticated architectures with faster processors, cache RAM, faster disk drives, and other benefits. Raw computing power can be the single most important consideration for the performance of your system.

Data Transfer Rates

A common benchmark for VXI computers is the Block Data Rate. This benchmark is easy for vendors to isolate and measure under ideal conditions. It is important to understand what Block Data Rate means to your application. Block Data Rate is the rate at which you can move a large block of data to or from memory on an ideal VXI device using back-to-back VXI transfers. It does not measure how fast the computer can process the blocks of data or store them to disk once they are moved, or whether your instruments themselves can actually match that data rate. Most applications are not limited by the Block Data Rate of the VXI interface hardware, but rather by the total time required to both move and handle the data, or by the rate at which the instruments themselves can generate or accept the data.

Block Data Rate is easy for vendors to specify, but often difficult for users to relate to overall system performance. It is only one of many elements that affect the actual throughput of your system. For example, Block Data Rate does not indicate the processing power of your computer or the performance of the instruments themselves. In addition, a benchmark for Block Data Rate does not measure how fast you can control instruments using VXI Word Serial Protocol or random VXI reads and writes. The speed for Word Serial communication and random VXI reads and writes is dependent on the speed of the processor and the particular VXI instruments.

Local Performance

The MXIbus does not degrade the performance of the devices connected to it. Each MXI device can operate internally at full speed in parallel with other MXI devices. Because MXIbus is a true system bus with multimaster arbitration, the only time MXI devices must synchronize their operation is when they perform transactions that map across the MXIbus. When one MXI device performs a read or write that maps to a remote MXI device, the MXI hardware on both devices interlocks the bus cycle across the MXIbus to accomplish the transfer.

MXI – An Open Standard

The MXIbus specification was developed by National Instruments and announced in April 1989 as an open industry standard. A *VXIplug&play* core technology, MXIbus has been endorsed by the entire *VXIplug&play* Systems Alliance, including Tektronix, Hewlett-Packard, Racal Instruments, and GenRad. Because MXI is an open standard documented with a comprehensive specification, anyone can develop products that will be integrated into a MXI controlled system.



34-02436-01
April 1996